

Konfliktne situacije – Student 3

Vukašin Jokić

1. Konkurentno odobravanje recepata

Scenario ponašanja aplikacije kod odobravanja recepata:

Medicinska sestra ima mogućnost odobravanja recepata koje doktor prepíše pacijentu tokom pregleda. Kada sestra odobri recept sistem beleži da je on overen kao i koja medicinska sestra ga je overila. Pošto mi želimo da našu aplikaciju koristi što veći broj korisnika može doći do seleće konfliktne situacije. Ako dve sestre odobre isti recept u isto vreme može doći do situacije da aplikacija prikaže da je recept uspešno overen od strane obe sestre iako je zapravo uspešno overen samo od jedne.

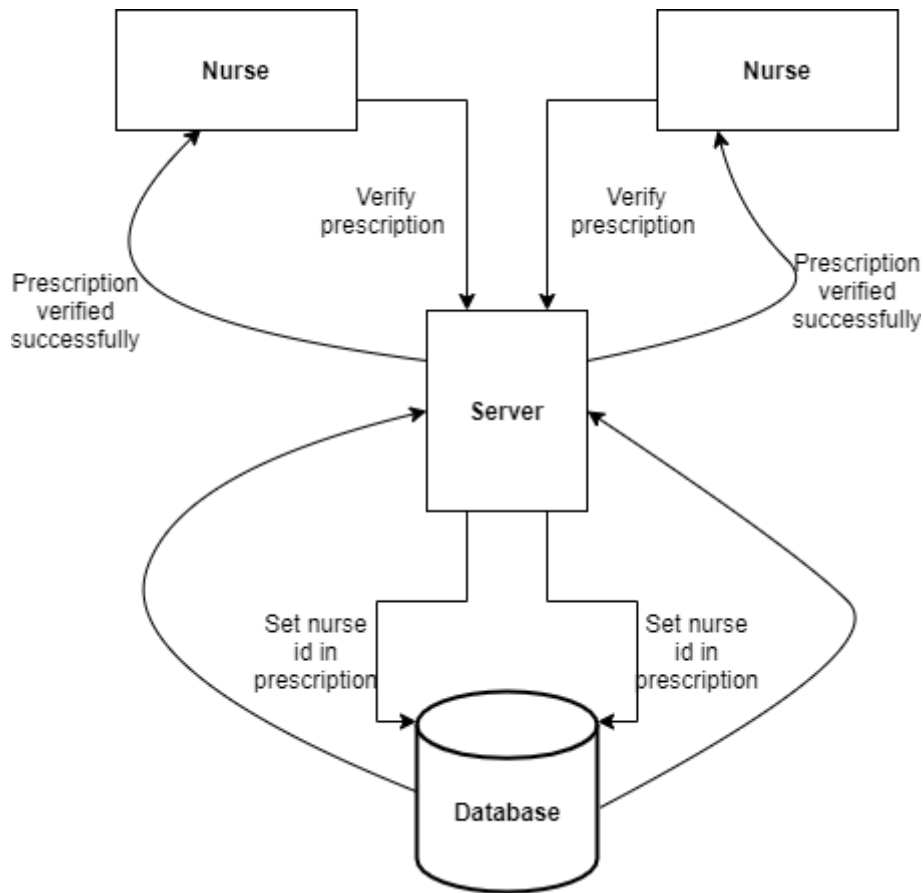


Figure 1 Šematski prikaz moguće konfliktne situacije kod overavanja recepata

Rešenje: Pomenuti problem je moguće rešiti upotrebom optimističkog zaključavanja. Potrebno je dodati novi atribut version, tipa long, u klasu Prescription sa anotacijom @Version. Ovo polje koristimo za proveru validnostu datog reda iz baze podataka. Tako da će se uspešno izvršiti prva transakcija dok

druga neće moći da se izvrši zbog uviđanja da je atribut *prescriptiona* promenjen u toku izvršavanja same transakcije.

2. Jedan lekar ne može istovremeno da bude prisutan na više različitih operacija

Scenario kada dolazi do konflikta je sledeći. Postoje više zahteva za pregled kod istog doktora u isto vreme. Dva admina klinike u isto vreme odobravaju te zahteve. Može se desiti da se kalendaru doktora postave isti datumi za pocetak operacije, što naravno ne bi trebalo da se desi.

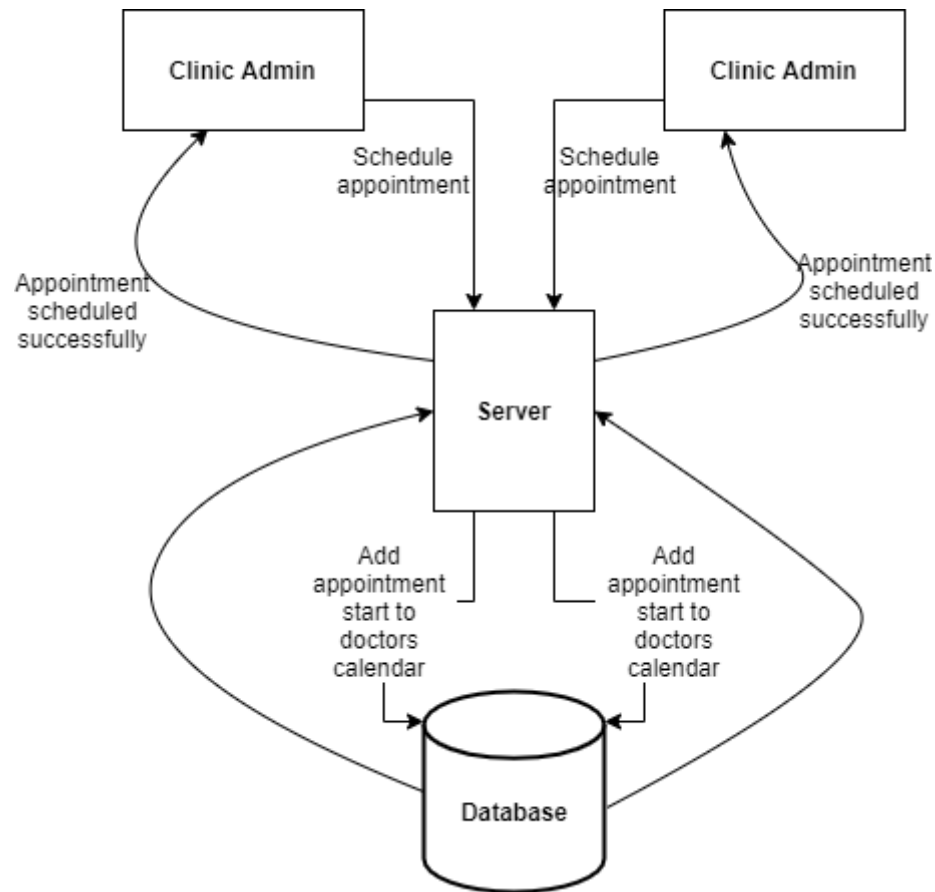


Figure 2 Šematski prikaz moguće konfliktne situacije pri odobravanju zahteva za pregled

Ovu situaciju možemo da rešimo uvođenjem *unique* ograničenja. Ovo ograničenje će obuhvatati kombinaciju polja *calendar_id* i *event_start_dates* što znači da se ne mogu naći dva pregleda sa istim vremenom početka u doktorovom kalendaru. U problemu koji smo opisali ovaj sistem bi vratio *unique constraint violation* grešku i time održao konzistentno stanje baze.